The Minimum-Cost Perfect Matching Problem

- Let $G = (V, E)$ be a bipartite graph such that the following conditions hold
  - V can be partitioned into two sets X and Y such that $|X| = |Y| = n$ and each edge in E has one endpoint in X and one endpoint in Y
  - G admits a perfect matching, i.e., a matching of cardinality n
  - Each edge e in E has a nonnegative cost c_e
- We define the cost of any subset E' of E as $\sum_{e \in E'} c_e$
- We wish to compute a minimum-cost perfect matching of G

An Iterative Framework

- We will compute a sequence of $n + 1$ matchings $M_0, \ldots, M_n$ where $|M_i| = i$
  - Thus the initial matching M_0 is the empty matching
- To obtain M_{i+1} from M_i, we will identify a suitable "augmenting path" P_i and set M_{i+1} to $M_i \oplus P_i$
  - P_i is a (simple) path in G of odd length, with one endpoint in X and one endpoint in Y
  - The edges of P_i alternate between edges in M_i and edges not in M_i
  - Neither endpoint is matched in M_i, so P_i begins and ends with an edge that is not in M_i
- When choosing the augmenting path in each iteration, we will also need to take edge costs into account

The Residual Graph G_M

- Suppose that after $i < n$ iterations, our current matching is M
- To identify a suitable augmenting path P for M, we construct the following "residual graph" G_M
  - The vertex set of G_M is $V \cup \{s, t\}$, where s and t are two new vertices called the source and sink, respectively
  - For each vertex x in X
    - If x is unmatched in M, there is a directed edge (s, x) in G_M with cost zero
    - For each edge $e = (x, y)$ in $E \setminus M$, G_M contains a directed edge from x to y with cost c_e
    - For each edge $e = (x, y)$ in M, G_M contains a directed edge from y to x with cost $-c_e$
  - For each unmatched vertex y in Y, there is a directed edge from y to t in G_M with cost zero

Augmenting Paths and the Residual Graph

- There is a one-to-one correspondence between (simple) directed s-t paths in G_M and augmenting paths in G with respect to matching M
  - Let P be an augmenting path in G with respect to M
  - P has an unmatched endpoint x in X and an unmatched endpoint y in Y
  - Let P' be the s-t path that starts with edge (s, x), follows the directed edges in G_M corresponding to P to go from x to y, and then uses edge (y, t) to get to t
    - Note that the edges in G_M corresponding to P are directed appropriately
  - Similarly, given a directed s-t path P' in G_M, we can obtain a corresponding augmenting path P in G with respect to M
    - Drop the first and last edges of P', and ignore the directions on the remaining edges

Achieving a Perfect Matching

- Lemma 1: If M is a matching of G with $|M| < n$, then there is an s-t path in G_M
  - Let M' be a perfect matching of G
    - Recall that we are assuming G has a perfect matching
  - The set of edges $M \oplus M'$ corresponds to a collection of vertex-disjoint alternating paths and cycles in G
  - Since $|M'| > |M|$, this collection includes an augmenting path P in G with respect to M
  - There is an s-t path P' in G_M corresponding to P

Refining our Iterative Framework

- Suppose our current matching is M with $|M| < n$
- We will identify a suitable s-t path P in G_M
- Let P' denote the corresponding augmenting path in G with respect to M
- We will update our matching M to $M' = M \oplus P'$
- The cost of M' is equal to the cost of M plus the cost of s-t path P
  - Each edge e in $P \setminus M$ contributes c_e to the cost of P'
  - Each edge e in $P \cap M$ contributes $-c_e$ to the cost of P'
- We will prove that if P is chosen to be a minimum-cost s-t path, the algorithm works

Alternating Cycles and the Residual Graph

- There is a one-to-one correspondence between (simple) directed cycles in G_M and alternating cycles in G with respect to M
  - Let C be an alternating cycle in G with respect to M
  - Let C' be the directed cycle in G_M consisting of the directed edges corresponding to the edges of C
    - Note that the edges in G_M corresponding to C are directed appropriately
  - Similarly, given a directed cycle C' in G_M, we can obtain a corresponding alternating cycle C in G with respect to M
    - Neither s nor t can appear on C' since s has indegree zero and t has outdegree zero
    - Ignore the edge directions on C' to get C

Characterizing Minimum-Cost Perfect Matchings

- Lemma 2: If M is a perfect matching of G and there is a negative-cost directed cycle C in G_M, then M is not a minimum-cost perfect matching
  - As we have seen, C corresponds to an alternating cycle in G with respect to M, call it C'
  - Let M' denote the perfect matching $M \oplus C'$
  - The cost of M' is equal to the cost of M plus the cost of C
  - Hence the cost of perfect matching M' is less than that of M

Characterizing Minimum-Cost Perfect Matchings (cont'd)

- Lemma 3: If M is a perfect matching of G and there is no negative-cost directed cycle in G_M, then M is a minimum-cost perfect matching
  - Let M' be a perfect matching of G
  - Thus $M' \oplus M$ corresponds to a collection $\mathcal{C}$ of vertex-disjoint alternating cycles in G with respect to M
  - $\mathcal{C}$ corresponds to a collection $\mathcal{C}'$ of vertex-disjoint directed cycles in G_M
  - The cost of M' is equal to the cost of M plus the sum of costs of the directed cycles in $\mathcal{C}'$
  - Since there is no negative-cost directed cycle in G_M, we conclude that the cost of M' is at least that of M

Vertex Prices and Reduced Costs

- Along with the matching M, we will maintain a "price" $p(v)$ for each vertex v in $V + s$
- We use the vertex prices to assign a "reduced cost" to each directed edge in G_M that is not incident on t, as follows
  - Each directed edge of the form (s, x) is assigned a reduced cost of $p(s) - p(x)$
  - For each edge $e = (x, y)$ in E where x is in X and y is in Y
    - If directed edge (x, y) belongs to G_M, it has reduced cost $c_e + p(x) - p(y)$
    - If directed edge (y, x) belongs to G_M, it has reduced cost $-c_e + p(y) - p(x)$

Nonnegative Reduced Costs: Benefit #1

- If we can choose the vertex prices so that all of the reduced costs are nonnegative, we gain in two ways
- First, we are assured that G_M does not contain any negative-cost directed cycles
  - Let C be a directed cycle in G_M
  - Vertex t (and also s) cannot appear on C, so every edge on C has a reduced cost
  - When we sum the reduced costs of the edges on C, the contributions of the vertex prices cancel out
  - Accordingly, the cost of C is equal to the reduced cost of C
  - Since the reduced costs are nonnegative, we conclude that the cost of C is nonnegative

Nonnegative Reduced Costs: Benefit #2

- Second, if the reduced costs are all nonnegative, then we can use Dijkstra's SSSP algorithm to compute a minimum-cost s-t path in G_M
  - For any vertex v in V, let $d_{p,M}(v)$ denote the minimum reduced cost of any s-v path in G_M
  - The minimum cost of any s-v path in G_M is thus $d_{p,M}(v) + p(v) - p(s)$
  - Let y minimize the expression $d_{p,M}(y) + p(y) - p(y)$ over all unmatched vertices y in Y, and let P be a minimum reduced cost s-y path in G_M
  - A minimum-cost s-t path in G_M is given by P plus (y, t)

Compatible Prices

- We say that vertex prices p are compatible with matching M if the following conditions are satisfied
  - We have $p(s) = 0$ and $p(x) = 0$ for all unmatched vertices x in X
    - Thus the reduced cost of any edge incident on s in G_M is zero
  - For any vertex x in X
    - The reduced cost $c_e + p(x) - p(y)$ of any directed edge (x, y) in G_M is nonnegative
    - The reduced cost $-c_e + p(y) - p(x)$ of any directed edge (y, x) in G_M is zero

Initialization of the Vertex Prices

- We wish to maintain the invariant that the vertex prices p are compatible with the current matching M
- The initial matching is the empty matching
- We initialize the price of each vertex in $V + s$ to zero
- The initial prices p and matching M are easily seen to be compatible

Updating the Vertex Prices

- Let M and p denote the matching and compatible vertex prices before a given iteration
- We have already seen how to update M to a suitable matching M'
- It remains to show how to update p to p' so that that M' and p' are compatible
- For each vertex u in $V + s$, we set $p'(u)$ to $p(u) + d_{p,M}(u)$
- It is easy to see that $p'(s) = 0$ and $p'(x) = 0$ for all unmatched vertices x in X
- Let $e = (x, y)$ be an (undirected) edge in E where x is in X and y is in Y
  - It remains to verify the compatibility condition for the corresponding directed edge in $G_{M'}$

Case 1: Edge $e = (x, y)$ belongs to M

- Claim: $c_e + p'(x) - p'(y) = 0$
  - Since the only edge entering x in G_M is $e = (y, x)$, we have $d_{p,M}(x) = d_{p,M}(y) - c_e + p(y) - p(x)$
  - Thus $c_e + p'(x) = d_{p,M}(y) + p(y) = p'(y)$, and the claim follows
- If (x, y) (resp., (y, x)) belongs to $G_{M'}$, the claim implies that the reduced cost of (x, y) (resp., (y, x)) with respect to p' is zero

Case 2: Edge $e = (x, y)$ belongs to $M' \setminus M$

- In this case, edge (y, x) belongs to $G_{M'}$, and we need to prove that $c_e + p'(x) - p'(y) = 0$
  - Thus edge (x, y) belongs to the augmenting path P
  - Since P corresponds to a shortest s-t path P' in G_M, we deduce that $d_{p,M}(y) = d_{p,M}(x) + c_e + p(x) - p(y)$
  - Thus $c_e + p'(x) = d_{p,M}(y) = p'(y)$, and the claim follows

Case 3: Edge $e = (x, y)$ belongs to $E \setminus (M \cup M')$

- In this case, directed edge (x, y) belongs to both G_M and $G_{M'}$
- Since (x, y) belongs to $G_{M'}$, we need to prove that $c_e + p'(x) - p'(y) \geq 0$
  - Since edge (x, y) belongs to G_M, we have $d_{p,M}(y) \leq d_{p,M}(x) + c_e + p(x) - p(y)$
  - Thus $d_{p,M}(y) \leq c_e + p'(x) - p(y)$, and the desired inequality follows